

Evolutionary Image Composition Using Feature Covariance Matrices

Aneta Neumann, Zygmunt L. Szpak, Wojciech Chojnacki, and Frank Neumann

School of Computer Science, The University of Adelaide, SA 5005, Australia

{aneta.neumann, zygmunt.szpak, wojciech.chojnacki,
frank.neumann}@adelaide.edu.au

Abstract

Evolutionary algorithms have recently been used to create a wide range of artistic work. In this paper, we propose a new approach for the composition of new images from existing ones, that retain some salient features of the original images. We introduce evolutionary algorithms that create new images based on a fitness function that incorporates feature covariance matrices associated with different parts of the images. This approach is very flexible in that it can work with a wide range of features and enables targeting specific regions in the images. For the creation of the new images, we propose a population-based evolutionary algorithm with mutation and crossover operators based on random walks. Our experimental results reveal a spectrum of aesthetically pleasing images that can be obtained with the aid of our evolutionary process.

Keywords: evolutionary algorithms, features, covariance matrices, image composition, digital art

1 Introduction

Evolutionary algorithms have been used in a wide range of areas to come up with novel solutions. This includes the classical task of computing high performing solutions for combinatorial optimisation problems, new designs in the area of engineering, as well as creative solutions in the areas of music and art [2, 15, 18, 22]. In the research area of evolutionary algorithms and art, the primary aim is to evolve artistic and creative outputs through an evolutionary process [1, 8, 17, 24]. The composition of images has gained some attention in the literature and mainly focuses on using textures. In

these works, genetic programming is used to synthesise procedural texture formulas [23]. Furthermore, genetic programming has been used to evolve functions constructing images based on aesthetic features [5, 6].

This paper presents an approach to carry out image composition directly based on two given images. Recently, evolutionary processes have been used to create artistic videos through evolutionary image transition. In evolutionary image transition, a source image is transferred into a target image by an evolutionary algorithm, with the images constructed during this process forming frames of a video sequence [19, 20]. Here, we propose a variant of evolutionary image transition. We use the evolutionary process to create a composition of two images S and T . The key element to achieve this goal is to use the concept of a so-called region covariance descriptor, which is well known and well studied in the field of computer vision. Region covariance descriptors can be used to describe an image based on various sets of features related to different regions of the image. Given two images S and T , we aim to evolve an image X which consists of a mixture of pixels from S and T and is “close” to both S and T in terms of covariance descriptors for both images. One of the main contributions of this paper is in finding a way to incorporate region covariance descriptors into an appropriate fitness function. We investigate the impact of different choices of image features on the aesthetic properties of images created via evolutionary image composition. We also explore the aesthetic impact of weights with which various region covariance descriptors can be combined together to form a fitness function.

The fitness function based on region covariance descriptors is minimised using an evolutionary algorithm which constructs sets of composed images. A key element in our evolutionary algorithm are random walk algorithms used previously in the context of evolutionary image transition. A random walk can be run as part of a mutation phase to obtain transitional images between S and T . In the case of crossover, a random walk can be used to generate a crossover mask which is then used to produce an offspring from two parent individuals. The random walk operator introduced in [20] depends on a parameter t_{\max} which determines how many steps a mutation operator performs. Here, with a view to developing an efficient process, we allow for self-adaptation of t_{\max} during the evolutionary computation. This permits increase of t_{\max} when the algorithm is making sufficient progress, and decrease of t_{\max} when the progress is low. As a result, our evolutionary algorithm adapts to the different stages of the optimisation process. Self-adaption is a key element of evolutionary algorithms for continuous optimisation problems. Recently, Doerr and Doerr [7] have modified this notion for the discrete case and shown that a cogent use of the modification significantly speeds up the optimisation for simple benchmark functions. We use the self-adaptation mechanism in such a fashion that t_{\max} decreases by a factor when an offspring is rejected, and increases by a factor when an offspring is accepted.

In our experimental investigations we explore how different parameters influence the artistic merit of the resulting images. We explore the role of image features, distances between covariance matrices and region weighting schemes. One of our key innovations is to utilise a measure of visual attention to promote the incorporation of salient regions from both images into the new synthesised images.

The paper is structured as follows. Section 2 details some prerequisites concerning a covariance descriptor. Section 3 presents a fitness function that we propose for image composition. Section 4 introduces an evolutionary algorithm including the mutation and crossover operators based on random walks. In Section 5, we present the results of our experimental investigations, and finally finish with a short discussion and some conclusions.

2 Region Covariance Descriptor

In computer vision various image descriptors have been proposed and used to capture essential characteristics of particular classes of images. Amongst these the *region covariance descriptor*, introduced by Tuzel et al. [26], has proved particularly useful for various computer vision tasks including object tracking [21], pedestrian detection [25], action recognition [9], and medical imaging [14]. The region covariance descriptor is constructed based on a feature mapping that associates with each pixel in the image a finite-dimensional vector of numerical features such as intensity, colour, gradients, or filter responses. Once a feature mapping is specified, any region in the image gives rise to the covariance matrix of the feature mapping restricted to this region. More formally, let $X = (X_{ij})$ be a colour image with a corresponding carrier $\Omega = \{(i, j) \mid 1 \leq i \leq m \text{ and } 1 \leq j \leq n\}$ of pixel locations. Each matrix entry X_{ij} is a triplet of integers representing colour information as coordinates in some colour space. Given a region of interest $\mathcal{R} \subset \Omega$ and a feature mapping $\phi: \Omega \rightarrow \mathbb{R}^p$, the corresponding *region covariance* matrix is given by

$$\Lambda_{\mathcal{R}} = \frac{1}{|\mathcal{R}| - 1} \sum_{(i,j) \in \mathcal{R}} (\phi(i, j) - \mu_{\mathcal{R}})(\phi(i, j) - \mu_{\mathcal{R}})^{\top},$$

where $\mu_{\mathcal{R}} = |\mathcal{R}|^{-1} \sum_{(i,j) \in \mathcal{R}} \phi(i, j)$ and $|\mathcal{R}|$ denotes the number of pixels in the region of interest. The matrix $\Lambda_{\mathcal{R}}$ describes the variations of the length- n feature vectors $\phi(i, j)$ as (i, j) varies over the region \mathcal{R} . An example feature mapping, used for human detection, is

$$\phi(i, j) = \left[i, j, I_{ij}, \left(\frac{\partial I}{\partial i}\right)_{ij}, \left(\frac{\partial I}{\partial j}\right)_{ij}, \left(\frac{\partial^2 I}{\partial i^2}\right)_{ij}, \left(\frac{\partial^2 I}{\partial j^2}\right)_{ij}, \sqrt{\left(\frac{\partial I}{\partial i}\right)_{ij}^2 + \left(\frac{\partial I}{\partial j}\right)_{ij}^2}, \tan^{-1} \left(\left| \left(\frac{\partial I}{\partial i}\right)_{ij} \right| / \left| \left(\frac{\partial I}{\partial j}\right)_{ij} \right| \right) \right]^{\top},$$

Table 1: Description of Potential Features for ϕ

	Notation	Description
ij	i	vertical spatial coordinate
	j	horizontal spatial coordinate
rgb	r	red channel
	g	green channel
	b	blue channel
∂	$ \frac{\partial I}{\partial i} $	magnitude of first-order partial derivative in horizontal direction
	$ \frac{\partial I}{\partial j} $	magnitude of first-order partial derivative in vertical direction
∂^2	$ \frac{\partial^2 I}{\partial i^2} $	magnitude of second-order partial derivative in horizontal direction
	$ \frac{\partial^2 I}{\partial j^2} $	magnitude of second-order partial derivative in vertical direction
	$ \frac{\partial^2 I}{\partial i \partial j} $	magnitude of second-order mixed partial derivative
$edge$	$\sqrt{(\frac{\partial I}{\partial i})^2 + (\frac{\partial I}{\partial j})^2}$	magnitude of edge response
	$\tan^{-1} \left(\frac{ \frac{\partial I}{\partial i} }{ \frac{\partial I}{\partial j} } \right)$	edge orientation
hsv	h	hue (HSV colour space)
	s	saturation (HSV colour space)
	v	value (HSV colour space)

where I_{ij} is the image intensity at (i, j) , $(\frac{\partial I}{\partial i})_{ij}$, $(\frac{\partial^2 I}{\partial i^2})_{ij}$, ... are intensity derivatives at (i, j) , and the last two terms are the magnitude of edge response and the edge orientation at (i, j) . In line with the convention adopted by MATLAB's `rgb2gray` function, the image intensity can be suitably defined by

$$I_{ij} = 0.2989X_{ij}^R + 0.5870X_{ij}^G + 0.1140X_{ij}^B,$$

where $X_{ij} = [X_{ij}^R, X_{ij}^G, X_{ij}^B]^\top$ is the decomposition of the colour vector X_{ij} into RGB components. In this work we explore several candidate feature mappings obtained by selecting component elements from the set of features described in Table 1.

There are several advantages of using covariance matrices as region descriptors. The feature mapping proposes a natural way of fusing multiple features which might be correlated. A single covariance matrix extracted from a region is usually enough to match the region in different views and poses. The noise corrupting individual samples are largely filtered out through averaging which is intrinsic to the process of covariance computation. The covariance descriptors are low dimensional—each matrix $\Lambda_{\mathcal{R}}$ has only $p(p+1)/2$ different entries (p is often less than 10), which is a number significantly smaller than the number of histogram bins (going into

hundreds) or of raw pixels (going into thousands) used by other descriptors. Moreover, each $\Lambda_{\mathcal{R}}$ does not preserve information regarding the ordering and the number of underlying grid points. This implies a certain degree of scale and rotation invariance over the regions in different images. However, it should be noted that this near invariance is largely reduced if the feature mapping contains explicit information regarding the orientation of points, such as the gradient of image intensity. The same is true for illumination.

Covariance matrices are positive-definite and one can speak about a distance between a pair of covariance matrices once a distance measure is defined between members of the set of all real positive-definite matrices. Let $Sym(p)$ denote the set of all $p \times p$ symmetric real matrices, and let $Sym_+(p)$ denote the subset of $Sym(p)$ comprised of all $p \times p$ positive-definite matrices in $Sym(p)$. $Sym_+(p)$ can be endowed with a variety of distance measures [12, 13]. In what follows we shall consider three specific distances. One is the *Euclidean* metric given by

$$\text{dist}_E(P, Q) = \|P - Q\|_F,$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and P and Q are members of $Sym_+(p)$. Another is the *Log-Euclidean* metric given by

$$\text{dist}_L(P, Q) = \|\log P - \log Q\|_F,$$

where \log denotes the principal matrix logarithm [3]. (For an invertible real matrix without eigenvalues on the negative real axis, there always exists a unique real matrix logarithm, called the principal logarithm, whose eigenvalues lie in the strip $\{z \in \mathbb{C} \mid -\pi < \text{Im} z < \pi\}$; cf. [10, Theorem 1.31].) Yet another distance measure is the *affine-invariant* metric given by

$$\text{dist}_A(P, Q) = \|\log(P^{-1}Q)\|_F = \left\| \log(P^{-1/2}QP^{-1/2}) \right\|_F$$

(cf. [16, Chap. XII]). The label “affine-invariant” reflects the fact that dist_A is invariant under each mapping of the form $P \mapsto APA^T$, where A is a real invertible matrix A ; that is,

$$\text{dist}_A(P, Q) = \text{dist}_A(APA^T, AQA^T)$$

for all $P, Q \in Sym_+(p)$ and all invertible $p \times p$ matrices A . The affine-invariant metric can alternatively be written as

$$\text{dist}_A(P, Q) = \left(\sum_{i=1}^p \log^2 \lambda_i(P^{-1}Q) \right)^{\frac{1}{2}}, \quad (1)$$

where $\lambda_i(P^{-1}Q)$, $1 \leq i \leq p$, are the eigenvalues of $P^{-1}Q$. As the matrix $P^{-1}Q$ is similar to the symmetric matrix $P^{-1/2}QP^{-1/2}$, the eigenvalues $\lambda_i(P^{-1}Q)$ are all positive and hence the right-hand side of (1) is well defined for all P and Q in $Sym_+(p)$.

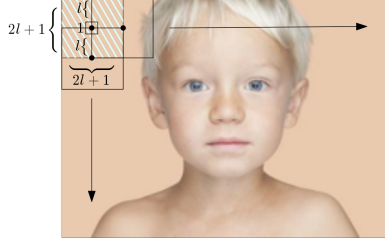


Figure 1: Illustration of overlapping square regions of interest.

3 Covariance-based Fitness Function

Given two images S and T both of size $m \times n$, we define the fitness of an image X of size $m \times n$ with respect to S and T . The fitness of the mixed image X is evaluated using region covariance descriptors that characterise how similar parts of the generated image are to both input images. The location and shape of the regions associated with the covariance descriptors can be chosen arbitrarily and are based on the personal preference of the artist. In this work we limit ourselves to square regions of interest arranged in a grid-like manner. In particular, we consider regions $\mathcal{R}_{(c,d)} = \{(i,j) \mid |i-c| \leq l, |j-d| \leq l\}$, which represent squares of size $(2l+1) \times (2l+1)$ centred at (c,d) , where (c,d) runs over a grid of pixels \mathcal{G} . The grid is defined as

$$\mathcal{G} = \left\{ (c,d) \left| \begin{array}{l} c = (l+1) + pl, \ p = 0, 1, \dots, \left\lfloor \frac{m-l}{l} \right\rfloor - 1 \\ d = (l+1) + ql, \ q = 0, 1, \dots, \left\lfloor \frac{n-l}{l} \right\rfloor - 1 \end{array} \right. \right\},$$

which results in half-overlapping square regions of interest (see Figure 1).

To measure the similarity between the generated image and the input images we propose the fitness function

$$f(X, S, T) = \sum_{(c,d) \in \mathcal{G}} \left(w_{(c,d)}^S \text{dist} \left(\Lambda_{\mathcal{R}_{(c,d)}}^X, \Lambda_{\mathcal{R}_{(c,d)}}^S \right) + w_{(c,d)}^T \text{dist} \left(\Lambda_{\mathcal{R}_{(c,d)}}^X, \Lambda_{\mathcal{R}_{(c,d)}}^T \right) \right),$$

where dist is one of the distance functions described in Section 2, and $w_{(c,d)}^S \in [0, 1]$ and $w_{(c,d)}^T \in [0, 1]$ are weights associated with the region $\mathcal{R}_{(c,d)}$ that can be used to emphasise the local contribution of the images S and T in the mixing process.

In addition to the fitness function, we put forward a constraint. Let $c_S(X) = |\{X_{ij} \mid X_{ij} = S_{ij}\}|$ be the number of pixels in X that are set to S and $c_T(X) = |\{X_{ij} \mid X_{ij} = T_{ij}\}|$ be the number of pixels where X and T

Algorithm 1 $(\mu + 1)$ GA for evolutionary image composition

Require: S and T are images

```
1: Initialise population  $\mathcal{P} = \{P_1, \dots, P_\mu\}$ 
2: while not termination condition do
3:   Select an individual  $P_i \in \mathcal{P}$  uniformly at random
4:   if  $\text{rand}() < p_c$  then ▷ Crossover
5:     Select  $P_j \in \mathcal{P} \setminus P_i$  uniformly at random
6:     if  $\text{rand}() < 0.5$  then ▷ See Section 4.2 for  $t_{\text{cr}}$ 
7:        $Y \leftarrow \text{RANDOMWALKMUTATION}(X, Z, t_{\text{cr}})$ 
8:     else
9:        $Y \leftarrow \text{RECTANGULARCROSSOVER}(P_i, P_j)$ 
10:     $P_i \leftarrow \text{SELECTION}(P_i, Y)$ 
11:  else ▷ Mutation
12:    if  $\text{rand}() < 0.5$  then
13:       $Y \leftarrow \text{RANDOMWALKMUTATION}(P_i, S, t_{\text{max}})$ 
14:    else
15:       $Y \leftarrow \text{RANDOMWALKMUTATION}(P_i, T, t_{\text{max}})$ 
16:     $P_i \leftarrow \text{SELECTION}(P_i, Y)$ 
17:    Adapt  $t_{\text{max}}$  ▷ See Section 4.1.
18: return  $\mathcal{P}$  ▷ Result is a population of evolved images.
```

agree. We minimise the fitness function f subject to the constraint

$$c(X) = |c_S(X) - c_T(X)| \leq B,$$

where B is an upper bound on how much the number of pixel in X from the two images S and T are allowed to differ.

4 Evolutionary Algorithm for Image Composition

In this section, we introduce our algorithm for creating artistic images using region covariance descriptors and an evolutionary image transition processes. Our method takes as input two images, $S = (S_{ij})$ and $T = (T_{ij})$ of size $m \times n$, and produces a new images $X = (X_{ij})$ of the same size, that is a mixture of the input images, i.e. $X_{ij} \in \{S_{ij}, T_{ij}\}$ for each $1 \leq i \leq m$ and each $1 \leq j \leq n$. The mixed images are generated using the genetic algorithm given in Algorithm 1. The algorithm uses a parent population of μ images and produces in each iteration one image Y by crossover with probability p_c or by mutation with probability $1 - p_c$. The initial population is a multi-set of images of the given images S and T . More precisely, each initial individual P_i in the initial population is chosen uniformly at random from $\{S, T\}$. In order to produce a diverse set of images, an offspring Y is only competing

Algorithm 2 RANDOMWALKMUTATION(X, Z, t_{\max})

Require: X and Z are images

- 1: $Y \leftarrow X$
 - 2: $Y_{ij} \in Y$ ▷ Choose starting pixel uniformly at random
 - 3: $Y_{ij} \leftarrow Z_{ij}$
 - 4: $t \leftarrow 1$
 - 5: **while** $t \leq t_{\max}$ **do**
 - 6: Choose $Y_{kl} \in N(Y_{ij})$ uniformly at random
 - 7: $i \leftarrow k, j \leftarrow l$ and $Y_{ij} \leftarrow Z_{ij}$
 - 8: $t \leftarrow t + 1$
 - 9: **return** Y ▷ Result is a mutated image.
-

against the first parent P_i for survival. This leads to a low selection process and produces a diverse set of images in the final population. We describe the different components of the genetic algorithm in the following section.

4.1 Self Adaptive Random Walk Mutation

Our genetic algorithm uses a variant of the random walk mutation introduced in [20]. The mutation operator is shown in Algorithm 2. Given an image Z the random walk starts at a random pixel of the current image X and produces an offspring Y from X by setting each visited pixel Y_{ij} to the value of Z_{ij} .

The random walk moves from a current pixel X_{ij} either left, right, up, or down to visit the next pixel and does this for t_{\max} steps. To formalise this, we define the neighbourhood $N(X_{ij})$ of X_{ij} as

$$N(X_{ij}) = \{X_{(i-1)j}, X_{(i+1)j}, X_{i(j-1)}, X_{i(j+1)}\}$$

and choose in each step an element of $N(X_{ij})$ uniformly at random. We use modulo the dimensions of the image which implies that the walk may wrap around the boundaries of the image.

Given a current image X , our genetic algorithm uses the random walk mutation to either paint all the visited pixels with the same values as in S or T . Whether to choose S or T is decided uniformly at random for each mutation operation. In this way, the algorithm is able to produce images that are mixtures of S and T and minimise the given fitness function. Each random walk mutation is run for t_{\max} steps.

Self Adaptation

When the algorithm successfully minimises the fitness function, we choose to increase the length of random walks by increasing t_{\max} . On the other hand, if progress can only be achieved by small random walk mutations, we

decrease t_{\max} . In particular, we employ the approach for adjusting discrete parameters recently introduced by Doerr and Doerr [7]. Their approach adapts the classical 1/5-rule adaptation for evolution strategies in [4] to the discrete setting. Our approach increases t_{\max} in the case of a success and decreases t_{\max} if the new offspring is not accepted. With the benefit of a self-adjustable mechanism, t_{\max} can take on values in $t_{\text{LB}} \leq t_{\max} \leq t_{\text{UB}}$, where t_{LB} is a lower bound on t_{\max} and t_{UB} is an upper bound on t_{\max} . This differs from the approach in [7], where the offspring population size is reduced in the case of a success and increased in the case of a failure.

For a successful mutation, we set

$$t_{\max} := \min \{F \cdot t_{\max}, t_{\text{UB}}\}$$

and for an unsuccessful mutation, we set

$$t_{\max} := \max \left\{ F^{-1/k} \cdot t_{\max}, t_{\text{LB}} \right\},$$

where $F > 1$ is a real value and $k \geq 1$ an integer which determines the adaptation scheme.

The scheme implements a $1/(k+1)$ -rule where F determines the size of the step changes dependent on the current value of t_{\max} . For our experimental investigations, we set $t_{\text{LB}} = 50$, $t_{\text{UB}} = 5000$, $F = 2$, $k = 8$ based on preliminary experimental investigations. Note that the use of t_{\max} in Algorithm 2 does not require it to be an integer. For a positive real value t_{\max} , the mutation operator will carry out $\lfloor t_{\max} \rfloor$ steps of the random walk.

4.2 Crossover

We consider two crossover operators which take two parents and produce one child.

The first crossover operator produces an offspring Y by taking the image of the first parent P_i and performing a random walk for t_{cr} steps on the image of the second parent P_j . This means that the offspring Y can be produced by calling `RANDOMWALKMUTATION(P_i, P_j, t_{cr})`. For our experimental investigations in this paper, we set $t_{\text{cr}} = 10000$, i.e. each random walk crossover mask is generated by a random walk consisting of 10000 steps. Note, that this value is independent of t_{\max} which is adapted during the mutation step. The reason is that we always want to make sure that crossover reduces an offspring which has sufficiently large parts from both parents.

The second crossover operator that we dub `RECTANGULARCROSSOVER` produces an offspring where all entries are copied from parent P_i except from a randomly specified rectangle R . The upper left point of the rectangle is chosen uniformly at random from the image. The width and the height of the image is chosen as a random integer in $\{1, \dots, m/10\}$ and $\{1, \dots, n/10\}$,

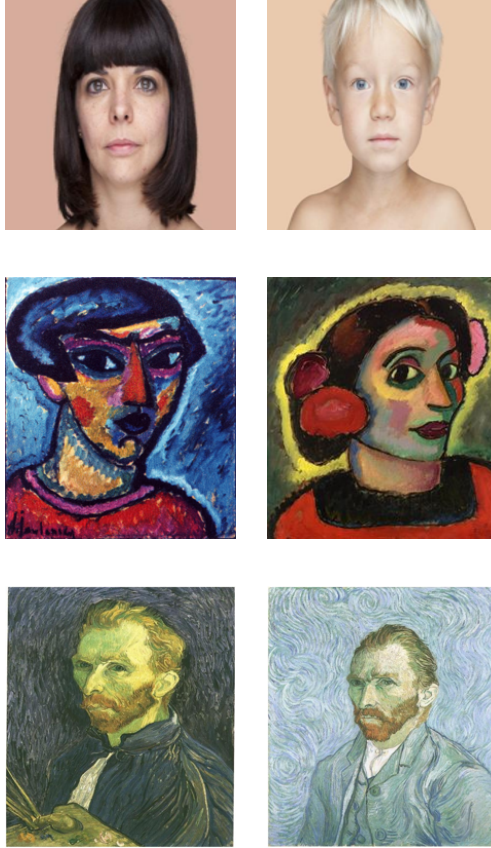


Figure 2: Pairs of images S (left column) and T (right column).

respectively. Values where the width and/or height would exceed the image boundaries are ignored.

Both crossover operators are focused on a local part of the target image. This facilitates the creation of offspring that have an identified part of the image of the second parent substituted into the image of the first parent. The hope is that the optimisation process benefits from this structured way of creating offspring.

5 Experiments

Our experiments were designed to meet a threefold objective: (1) we wished to investigate the impact of different region covariance features on the resulting images; (2) we wanted to discover how different weighting schemes for covariance matrices influence the results; and (3) we wished to understand the influence that the distance measures have on the final results.



Figure 3: Image composition with different features. Rows 1, 2 and 3 correspond to Feature Sets 1, 2 and 3, respectively. Note the structure that emerges with the first feature set.

The pairs of images S and T for all of our experiments are depicted in Figure 2. For each setting investigated in this section, we ran our genetic algorithm for 2000 generations with a population size of $\mu = 4$ and crossover probability $p_c = 0.2$.

5.1 Impact of Features

Our first set of experiments were focused on characterising how the choice of features may influence the resulting image. With reference to Table 1 we chose 3 sets of features as follows:

$$\text{Set 1: } \left[i, j, r, g, b, \sqrt{\left(\frac{\partial I}{\partial i}\right)^2 + \left(\frac{\partial I}{\partial j}\right)^2}, \tan^{-1} \left(\left| \frac{\partial I}{\partial i} \right| / \left| \frac{\partial I}{\partial j} \right| \right) \right]^T ;$$

$$\text{Set 2: } [i, j, h, s, v]^T ;$$

$$\text{Set 3: } \left[h, s, v, \sqrt{\left(\frac{\partial I}{\partial i}\right)^2 + \left(\frac{\partial I}{\partial j}\right)^2}, \tan^{-1} \left(\left| \frac{\partial I}{\partial i} \right| / \left| \frac{\partial I}{\partial j} \right| \right) \right]^T .$$

For all experiments in this section we considered a grid of equispaced square region covariance descriptors ($l = 25$ pixels). We set both sets of

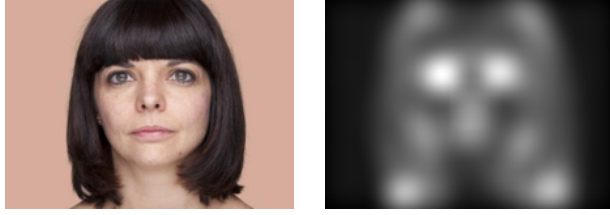


Figure 4: An image and its corresponding saliency mask.

weights $w_{(c,d)}^S$ and $w_{(c,d)}^T$ to 0.5, and utilised the Euclidean distance as our measure of similarity between covariance matrices.

The results in Figure 3 demonstrate that the three populations evolve differently depending on the feature set. Feature Set 1 produced the most visually pleasing results, with the composite image incorporating facial features from both subjects. We attribute this to the correlations that are captured between pixel locations and edge features (magnitude and orientation) in the first feature set, and not in the other two feature sets.

5.2 Impact of Different Weightings

In the second set of experiments we explored the consequence of using different weights for the region covariance matrices. We limited ourselves to two weighting schemes: uniform and saliency-based. In the uniform regime all of the weights $w_{(c,d)}^S$ associated with S were set to the same value, and, similarly, all the weights $w_{(c,d)}^T$ associated with T were also fixed. In particular, for simulations 1, 2 and 3, we set $w_{(c,d)}^S$ to 0.2, 0.5 and 0.75 and $w_{(c,d)}^T$ to 0.75, 0.5 and 0.25, respectively.

In contrast, for a simulation with saliency-based weighting, we utilised the image saliency algorithm of [11] to assign weights for each region covariance descriptor. The purpose of the saliency algorithm is to predict human fixation points, which in turn are used as a measure of visual attention. The saliency algorithm takes as input an image, and produces a new image of the same size which can be interpreted as a probability map designating which regions of an image a human would pay most of their attention to (see Figure 4). We ran the saliency algorithm on both S and T . The weights for $w_{(c,d)}^S$ and $w_{(c,d)}^T$ were taken from the saliency maps associated with S and T , respectively.

For all experiments in this section we used Feature Set 1 and employed a grid of equispaced square region covariance descriptors ($l = 20$ pixels). We measured the similarity between covariance matrices using the Log-Euclidean distance.

The results of our experiment are depicted in Figure 5. As expected,



Figure 5: Image composition with different covariance weighting schemes. Rows 1, 2 and 3 correspond to $w_{(c,d)}^S$ set to 0.25, 0.5 and 0.75 and $w_{(c,d)}^T$ set to 0.75, 0.5 and 0.25, respectively. In the last row the weights were set using an image saliency algorithm. The saliency algorithm strikes a consistent balance between notable regions in both images.

when uniform weights are used the global relative contribution of the image S can be traded-off against the global relative contribution of the image T . The benefit of the saliency-based weighting scheme is that it can automatically produce visually pleasing results that strike a balance between notable regions in both images.

5.3 Impact of Distance Measures

In the third and final set of experiments we investigated the utility of different metrics for covariance matrices. Once again, a notable structure emerges.

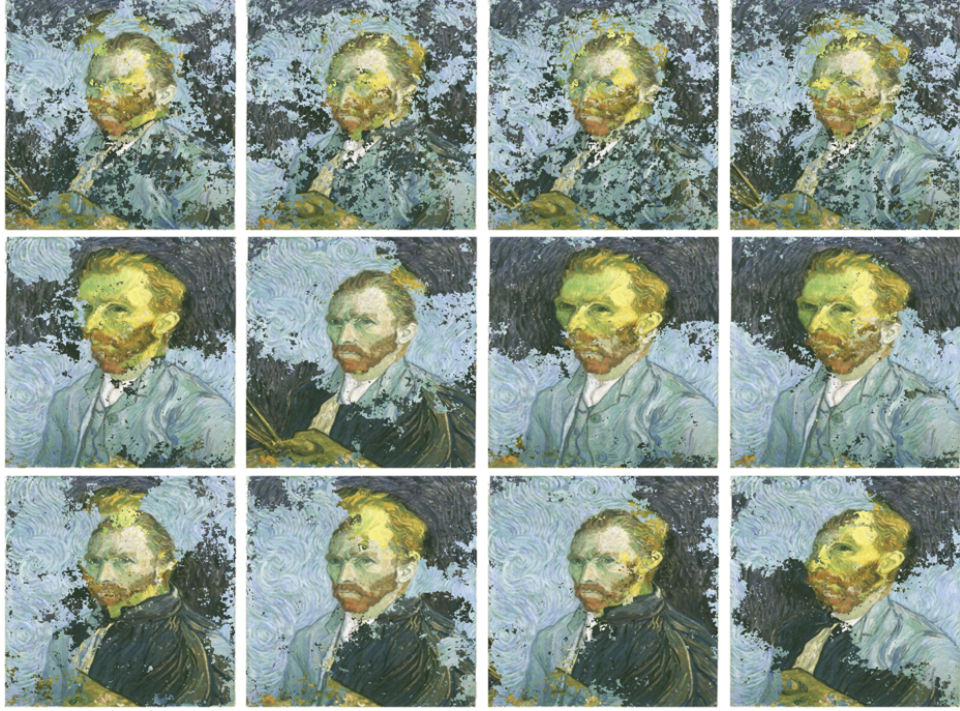


Figure 6: Image composition with different covariance distances. Rows 1, 2 and 3 correspond to distance metrics dist_E , dist_A and dist_L , respectively. Note how the Euclidean distance yields patchy results, whereas the other distance measures result in more structured images.

The Euclidean distance measure produced inferior results, with the images taking on a very patchy and noisy appearance. In contrast, the images generated using the Log-Euclidean or affine-invariant distance were much more cohesive and interesting. The results presented in Figure 6 are based on grid of equispaced square region covariance descriptors ($l = 20$ pixels) using saliency-based weights and Feature Set 1.

6 Discussion

Our experimental examinations suggest that the best results are obtained when Feature Set 1 is used in conjunction with saliency-based weighting and a Log-Euclidean distance measure. To verify this hypothesis, we deployed our genetic algorithm one more time with these parameters on all three image pairs. The results presented in Figure 7 confirm our hypothesis. In all cases, our algorithm produced a population of new aesthetic images that incorporate pertinent regions from both input images.



Figure 7: Image composition using Feature Set 1 with saliency-based weighting and the Log-Euclidean distance measure.

7 Conclusion

We have introduced a new approach of image composition based on feature covariance matrices. This approach facilitates the composition of novel artistic images. We have introduced a genetic algorithm evolving a set of images by crossover and mutation operators based on random walk. For our experimental investigations, we have considered different pairs of images and compared the final populations showing the resulting images with respect to different parameters.

Acknowledgements

This work has been supported through Australian Research Council (ARC) grants DP140103400 and LE160100090.

References

- [1] M. M. al-Rifaie and J. M. Bishop. 2013. Swarmic Paintings and Colour Attention. In *Proc. 2nd Int. Conf. Evolutionary and Biologically In-*

- spired Music, Sound, Art and Design (EvoMUSART '13)*. Lecture Notes in Computer Science, Vol. 7834. Springer, Berlin - Heidelberg, 97–108.
- [2] R. F. Antunes, F. F. Leymarie, and W. H. Latham. 2015. On writing and reading artistic computational ecosystems. *Artificial Life* 21, 3 (2015), 320–331.
 - [3] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. 2006. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine* 56, 2 (2006), 411–421.
 - [4] A. Auger. 2009. Benchmarking the (1+1)-ES with one-fifth success rule on the BBOB-2009 noisy testbed. In *Proc. ACM Conf. Genetic and Evolutionary Computation (GECCO '09)*. ACM, New York, NY, USA, 2453–2458.
 - [5] E. den Heijer and A. E. Eiben. 2010. Using aesthetic measures to evolve art. In *Proc. IEEE Congress on Evolutionary Computation (CEC '10)*. IEEE, Los Alamitos, CA, Article C-7184, 8 pages.
 - [6] E. den Heijer and A. E. Eiben. 2014. Investigating aesthetic measures for unsupervised evolutionary art. *Swarm and Evolutionary Computation* 16 (2014), 52–68.
 - [7] B. Doerr and C. Doerr. 2015. Optimal parameter choices through self-adjustment: applying the 1/5-th rule in discrete settings. In *Proc. ACM Conf. Genetic and Evolutionary Computation (GECCO '15)*. ACM, New York, NY, USA, 1335–1342.
 - [8] G. Greenfield. 2015. Avoidance drawings evolved using virtual drawing robots. In *Proc. 4th Int. Conf. Evolutionary and Biologically Inspired Music, Sound, Art and Design (EvoMUSART '15)*. Lecture Notes in Computer Science, Vol. 9027. Springer, Cham, 78–88.
 - [9] K. Guo, P. Ishwar, and J. Konrad. 2010. Action recognition using sparse representation on covariance manifolds of optical flow. In *Proc. IEEE Conf. Advanced Video and Signal Based Surveillance (AVSS '10)*. IEEE, Los Alamitos, CA, 188–195.
 - [10] N. J. Higham. 2008. *Functions of Matrices: Theory and Computation*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.
 - [11] X. Hou, J. Harel, and C. Koch. 2012. Image signature: highlighting sparse salient regions. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 1 (2012), 194–201.

- [12] S. Jayasumana, R. Hartley, M. Salzmann, Hongdong Li, and M. Harandi. 2013. Kernel methods on the Riemannian manifold of symmetric positive definite matrices. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR '13)*. IEEE, Los Alamitos, CA, 73–80.
- [13] S. Jayasumana, R. Hartley, M. Salzmann, Hongdong Li, and M. Harandi. 2015. Kernel methods on the Riemannian manifolds with Gaussian RBF kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 12 (2015), 2464–2477.
- [14] A. M. Khan, K. Sirinukunwattana, and N. Rajpoot. 2014. Geodesic geometric mean of regional covariance descriptors as an image-level descriptor for nuclear atypia grading in breast histology images. In *Proc. 5th Int. Workshop on Machine Learning in Medical Imaging (MLMI '14)*. Lecture Notes in Computer Science, Vol. 8679. Springer, Cham, 101–108.
- [15] N. Lambert, W. H. Latham, and F. F. Leymarie. 2013. The emergence and growth of evolutionary art: 1980-1993. In *Proc. ACM SIGGRAPH Art Gallery (SIGGRAPH '13)*. ACM, New York, NY, USA, 367–375.
- [16] S. Lang. 1999. *Fundamentals of Differential Geometry*. Springer, New York.
- [17] P. Machado and J. Correia. 2014. Semantic aware methods for evolutionary art. In *Proc. ACM Conf. Genetic and Evolutionary Computation (GECCO '14)*. ACM, New York, NY, USA, 301–308.
- [18] J. McCormack and M. d’Inverno (Eds.). 2012. *Computers and Creativity*. Springer, Berlin - Heidelberg.
- [19] A. Neumann, B. Alexander, and F. Neumann. 2016. The evolutionary process of image transition in conjunction with box and strip mutation. In *Proc. 23rd Int. Conf. on Neural Information Processing (ICONIP '16)*. Lecture Notes in Computer Science, Vol. 9949. Springer, Cham, 261–268.
- [20] A. Neumann, B. Alexander, and F. Neumann. 2017. Evolutionary image transition using random walks. In *Proc. 6th Int. Conf. Evolutionary and Biologically Inspired Music, Sound, Art and Design (EvoMUSART '17)*. Springer, Cham. To appear.
- [21] F. Porikli, O. Tuzel, and P. Meer. 2006. Covariance tracking using model update based on Lie algebra. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR '06)*. IEEE, Los Alamitos, CA, 728–735.

- [22] J. Romero and P. Machado (Eds.). 2008. *The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music*. Springer, Berlin - Heidelberg.
- [23] B. J. Ross, W. Ralph, and H. Zong. 2006. Evolutionary image synthesis using a model of aesthetics. In *Proc. IEEE Congress on Evolutionary Computation (CEC '06)*. IEEE, Los Alamitos, CA, 1087–1094.
- [24] K. Trist, V. Ciesielski, and P. Barile. 2011. An artist’s experience in using an evolutionary algorithm to produce an animated artwork. *Int. J. Arts and Technology* 4, 2 (2011), 155–167.
- [25] O. Tuzel, F. Porikli, and P. Meer. 2008. Pedestrian detection via classification on Riemannian manifolds. *IEEE Trans. Pattern Anal. Mach. Intell.* 30, 10 (2008), 1713–1727.
- [26] O. Tuzel, F. M. Porikli, and P. Meer. 2006. Region covariance: a fast descriptor for detection and classification. In *Proc. 9th European Conf. Computer Vision (ECCV '06)*. Lecture Notes in Computer Science, Vol. 3952. Springer, Berlin - Heidelberg, 589–600.